

## Step1 : Setting database containers and Azure data bricks

This screenshot shows the Microsoft Azure portal interface for a resource named 'databricks-ws'. The left sidebar contains navigation links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Monitoring, Automation, and Help. The main content area displays the 'Overview' tab with a search bar and a 'Delete' button. Below this, the 'Essentials' section provides key information: Status (Active), Resource group (databricks-rg), Location (East US), Subscription (Azure for Students), and Subscription ID (884151b7-396c-4e75-9278-e15b2130e2e2). It also lists the Managed Resource Group, URL, and Pricing Tier. A large red Databricks logo is centered below the essentials. At the bottom, there is a 'Launch Workspace' button and a grid of links for Documentation, Getting Started, Import Data from File, Import Data from Azure Storage, Notebook, Admin Guide, and Link Azure ML workspace.

This screenshot shows the Microsoft Azure portal interface for a deployment named 'santoshdb8\_1733637984046'. The left sidebar shows navigation links for Overview, Inputs, Outputs, and Template. The main content area displays the 'Overview' tab with a search bar and buttons for Delete, Cancel, Redeploy, Download, and Refresh. The status 'Deployment is in progress' is shown. Below this, the 'Deployment details' section provides information: Deployment name (santoshdb8\_1733637984046), Start time (12/8/2024, 11:36:41 AM), Subscription (Azure for Students), Resource group (databricks-rg), and Correlation ID. A table with columns Resource, Type, Status, and Operation details is shown, but it contains no results. On the right, there are links for Microsoft Defender for Cloud, Free Microsoft tutorials, and Work with an expert.

This screenshot shows the Databricks workspace configuration page for a cluster named 'eti-cluster'. The left sidebar contains navigation links for New, Workspace, Recents, Catalog, Workflows, Compute, SQL, SQL Editor, Queries, Dashboards, Genie, Alerts, Query History, SQL Warehouses, Data Engineering, Job Runs, Data Ingestion, Delta Live Tables, Machine Learning, Playground, Experiments, Features, Models, Serving, Marketplace, and Partner Connect. The main content area displays the 'Configuration' tab for the 'eti-cluster'. It shows the Policy (Unrestricted), Access mode (Single user access), and Performance settings (Databricks Runtime Version 11.3 LTS, Use Photon Acceleration, Worker type Standard\_D4ds\_v5, Driver type Standard\_D4ds\_v5). A 'Summary' panel on the right provides a high-level overview of the cluster configuration, including 2-8 Workers, 32-128 GB Memory, 8-32 Cores, 1 Driver, 16 GB Memory, 4 Cores, Runtime 11.3.x-scala2.12, and 6-18 DBU/h.

Step 2: Run the python code in Azure blob notebook (Load large dataset, preprocess, Implement ml ):

```
12 (In [1]:)
Output Terminal Debug Console

DataFrame loaded successfully!
  product_id product_name ... customer_location target_column
0      2001   Smartwatch ...           Seattle             1
1      2002   Air Fryer ...           Chicago             0
2      2003   Bookshelf ...           Boston             0
3      2004    Tablet ...        New York             1
4      2005   Microwave ...        California             1

[5 rows x 9 columns]
```

```
12 (In [1]:)
Output Terminal Debug Console

new_data_df: pyspark.sql.dataframe.DataFrame = [feature1: double, feature2: double ... 3 more fields]
prediction_df: pyspark.sql.dataframe.DataFrame = [prediction: double]
Mean Squared Error (MSE): 0.09963745098447913
Model uploaded successfully to Azure Blob Storage as linear_regression_model.pkl
+-----+
| prediction |
+-----+
| 0.7039318272466477 |
| 0.6666204388630401 |
| 0.707304079479652 |
| 0.716115258049575 |
+-----+

Predictions saved to Azure Blob Storage as predictions.csv
```

```
12 X = np.random.rand(100, 5) # 100 samples, 5 features
13 y = np.random.rand(100) # 100 target values

Output Terminal Debug Console

Mean Squared Error (MSE): 0.07287475316359711
Model successfully uploaded to Azure Blob Storage as linear_regression_model.pkl
```